NAME: LAKSHYA KANT , REG. NO.: 24BCE10549
SUBJECT: DSA, ASSIGNMENT

CODE :

```c
/*--Details--
Name: Lakshya Kant
Reg. No.: 24BCE10549
Date: 05-09-2025
*/


/*
The MOST SUITABLE Data Structure for this task is Circular Queue.
Justification:
1.Because they need to be cycled repeatedly in the same order, just like a
circle with 4 points.
2.No of elements to be cycled is fixed.
3.Other Data Structures such as a simple array or linked list are not
efficient for this task, as there is no inbuilt concept of circularity in
arrays, and linked list has variable size, which isn't required here.
*/


#include <stdio.h>
#include <stdlib.h>
#define size 4   // number of directions

// Circular Queue structure
struct circularQueue{
    char* arr[size];
    int front, rear;
};

// Initialize queue
void initQueue(struct circularQueue* q) {
    q->front = -1;
    q->rear = -1;
}
// Enqueue operation
void enqueue(struct circularQueue* q, char* direction) {
    if ((q->rear + 1) % size == q->front) {
```

```c
            printf("Queue is Full! Cannot add %s\n", direction);
            return;
        }
    if (q->front == -1) {
        q->front = q->rear = 0;
    } else {
        q->rear = (q->rear + 1) % size;
    }
    q->arr[q->rear] = direction;
}

// dequeue operation
char* dequeue(struct circularQueue* q) {
    if (q->front == -1) {
        printf("Queue is Empty!\n");
        return NULL;
    }
    char* item = q->arr[q->front];
    if (q->front == q->rear) {
        q->front = q->rear = -1; // reset when last element removed
    } else {
        q->front = (q->front + 1) % size;
    }
    return item;
}

// simulate traffic controller
void runTrafficController(struct circularQueue* q, int cycles) {
    for (int i = 0; i < cycles; i++) {
        char* dir = dequeue(q);
        if (dir) {
            printf("Green Light: %s\n", dir);
            enqueue(q, dir); // put it back for circular effect
        }
    }
}

int main() {
    struct circularQueue q;
    initQueue(&q);
```

```c
    // Insert directions into queue
    enqueue(&q, "North");
    enqueue(&q, "East");
    enqueue(&q, "South");
    enqueue(&q, "West");

    int n;
    printf("Enter number of cycles to simulate: ");
    scanf("%d", &n);

    runTrafficController(&q, n);

    return 0;
}
```